and to which a claim of copyright protection is made. The owner has no objection to the facsimile reproduction by anyone of the patent disclosure, as it appears in the Patent and Trademark Office patent files or records, but otherwise reserves all copyrights and similar rights whatsoever.

The present invention is related to inter-process communication and the representation of data facilitating such inter-process communication for such applications as testing. Although the present invention will be described with reference to certain specific implemented embodiments and examples of such which use the data structure and formats to be described here for the application of testing, it can be appreciated by one skilled in the art that such self-describing data formats may be used for any number of applications as desired by implementors of this invention. Other advantages and applications for implemented embodiments will become apparent, to one skilled in the art, and may be practiced without these specific details without departing from the overall spirit and scope of the present invention.

The present invention is implemented as a series of data structures and accompanying instructions implemented in a computer program which is operative within a computer system. Such data structures may be created in a computer system as illustrated in the block diagram of FIG. **1**.

Referring to FIG. **1**, a system upon which one implementation of the present invention is implemented is shown as **100**. **100** comprises a bus or other communication means **101** for communicating information, and a processing means **102** coupled with bus **101** for processing information. System **100** further comprises a random access memory (RAM) or other volatile storage device **104** (referred to as main memory), coupled to bus **101** for storing information and instructions to be executed by processor **102**. Main memory **104** also may be used for storing temporary variables or other intermediate information during execution of instructions by processor **102**. System **100** also comprises a read only memory (ROM) and/or other static storage device **106** coupled to bus **101** for storing static information and instructions for processor **102**, and a data storage device **107** such as a magnetic disk or optical disk and its corresponding disk drive. Data storage device **107** is coupled to bus **101** for storing information and instructions. This may be used for storage of the databases to be described here which maintain information about currently defined problem descriptions using commercially available software products.

System **100** may further be coupled to a display device **121**, such as a cathode ray tube (CRT) or liquid crystal display (LCD) coupled to bus **101** for displaying information to a computer user. Such a display **121** may further be coupled to bus **101** via a frame buffer **110**, which information such as a single or multiple frames or images for display upon display device **121**. An alphanumeric input device **122**, including alphanumeric and other keys, may also be coupled to bus **101** for communicating information and command selections to processor **102**. An additional user input device is cursor control **123**, such as a mouse, a trackball, stylus, or cursor direction keys, coupled to bus **101** for communicating direction information and command selections to processor **102**, and for controlling cursor movement on display **121**.

Note, also, that any or all of the components of system **100** and associated hardware may be used in various embodiments, however, it can be appreciated that any configuration of the system may be used for various purposes according to the particular implementation.

In one embodiment, system **100** is one of the Sun Microsystems® brand family of workstations such as the SPARCstation workstations manufactured by Sun Microsystems® of Mountain View, Calif. Processor **102** may be one of the SPARC brand microprocessors manufactured by Sun Microsystems® of Mountain View, Calif. (Sun Microsystems® of Mountain View, Calif.).

Note that the following discussion of various embodiments discussed herein will refer specifically to a series of routines which are generated in a high-level programming language (e.g., the C language) and compiled, linked, and then run as object code in system **100** during run-time, for example by the SPARCompiler available from Sun Microsystems® of Mountain View, Calif. Specifically, the present invention is operative in conjunction with various software libraries, such as the Solaris® threads package available from SunSoft, Inc. of Mountain View, Calif. (Sun, Sun Microsystems and Solaris are trademarks of Sun Microsystems of Mountain View, Calif. SPARC and SPARC station are trademarks of SPARC International, Inc. and are licensed exclusively to Sun Microsystems). It can be appreciated by one skilled in the art, however, that the following methods and apparatus may be implemented in special purpose hardware devices, such as discrete logic devices, large scale integrated circuits (LSI's), application-specific integrated circuits (ASIC's), or other specialized hardware. The description here has equal application to apparatus having similar function.

The present invention implements a system wherein the application programmer generating the program under test specifies "probes" in the source code of the program so that the probes may later be activated at system run-time, depending upon certain parameters specified during the invocation of the executable program, causing the probes to generate records which may be examined by a second application program or user (e.g., a test engineer). In this manner, application programs under test may provide information regarding the data that they are processing, including, descriptive information which is included in the records which are generated from the application program under test itself. The details of this will be discussed in more detail below, however, the operation of such a mechanism is illustrated with reference to FIG. **2**.

200 of FIG. **2** illustrates an application program **210** typically a program under test which may be invoked with certain probe parameters **201**. This may include, such arguments specified by a test engineer **202** or other mechanism which specifies certain probes or groups of probes which will be activated upon invocation of the application program. The application program then generates probe events **220**, which are placed into a trace file **221** which contains trace records (to be described below) which may indicate the status of various probes which have been inserted into the application program. By detection that the probe parameters **201** have specified that certain probes or groups of probes should be activated, then probe events **220** placed into the trace file **221** is generated which may be examined by a second program **230**, typically a test suite or other application program which can receive and process the probe events **220** via the trace file **221**. The trace file **221** contain references to records which contain the data themselves from within the application program **210**, along with some identifying information. The generation and placement of records into a file such as trace file **221** is provided by the memory mapping (mmap) facilities of the Sun Operating System which is used in implemented embodiments such that the virtual memory requested of records generated using implemented embodiments is captured as a memory image in the file system of computer system **100**. In other computer